Report
Programming Class

Programming Class 24/25

Bachelor program UXD, THUAS

Student ID: 24097993

Kerr Beeldens

Picture by Albert Stoynov on Unsplash

X

Test

Contents

C1

Chapter 1

Introduction

This report contains my work from the programming class bootcamp. The report is split up into 6 chapters, with the first chapter (this one) being a short introduction and each of the following chapters is about 1 day of the bootcamp. These chapters are split up into a section with a summary of the topics of that day and a section that explains what the assignment was, what I tried to achieve and how you applied the topics from the "listen" segment. The final chapter is a reflection on the bootcamp as a whole.

The .zip file that contains this report also contains the source code for each of the 5 assignments. Appendix A1 through A5 also contain this source code of these assignments and I occasionally explain or show parts of the code throughout the document as well.

Link to GitHub

I am fully aware that no links are allowed and the .zip file is intended to stand on its own, but in case it is easier, the full contents are also available on GitHub through the following link:

https://github.com/KerrBeeldens/programming_class_24_25

Chapter 2

Day 1: shapes, colors & variables

2.1 Summary of topics

During this lecture the layout of the PDE was explained. A basic program to draw a line was shown. Here, it was showed that one can draw basic shapes in processing using a variety of functions. These functions have a number of parameters. These are things like numbers, strings or booleans that change the behavior of the function. Examples are:

Example 1

```
println("Hello"); // print "Hello" in the console
size(400, 400); // set the screen size to 400 by 400 pixels
point(50, 90); // draw a point at x = 50, y = 90
```

It was also explained that by using two slashes, you can create comments, which help explain code to others (like I did above). Furthermore, the coordinate system of the screen (going from left to right and from above to below) and the drawing order was shown. Colors of objects can be changed, by using:

Example 2

```
fill(255, 0, 0); // set the fill color to red
background(0, 255, 0); // set the background to green
stroke(0, 0, 255, 127); // set the stroke color to translucent blue
```

The 4 parameters representing the R, G, B and Alpha values.
Next, variables where explained. These are small pieces of data that can represent numbers, letters, words, colors booleans etc. They are created using a type specifier like `int` or `boolean` followed by a name. Naming variables is done using camelCase. One can then use this name to assign data to the variable and then use the variable like:

Example 3

```
int a = 20; // declare a variable a and assign 20 to it
point(a, a); // use a to place a point at 20, 20
```

Processing also has build-in variables, like `width` and `height`.
Then, some basic mathematics where shown. You can add values or variables together using "+", subtract them using "-", multiply using "*" and divide using "/". It is important to keep track of the data types when doing this. It is also possible to change the datatype. For example, changing a value to an integer is possible using `int()`.

Finally the `random()` function was explained. It returns a float between 0 and the given number or when two numbers are given, between the two numbers, like this:

Example 4

```
float randomNumber = random(20); // return a random float from 0-20
int randomNumber = int(random(5, 20)); // return a random int from 5-19
```

2.2 Challenge description: name

C3

Chapter 3

Day 2: flow & conditionals

3.1 Summary of topics

3.2 Challenge description: name

C4

Chapter 4

Day 3: conditionals & interactivity

4.1 Summary of topics

4.2 Challenge description: name

C5

Chapter 5

Day 4: loops & functions

5.1 Summary of topics

5.2 Challenge description: name

C6

Chapter 6

Day 5: arrays

6.1 Summary of topics

6.2 Challenge description: name

C7

Chapter 7

Reflection

Appendix A

Source code

A.1 Challenge 1: half-tone

challenge_1_halftone.pde

```
void setup() {
  // Initialize the window
  size(1920, 1080);
  background(255);
  noStroke();
  fill(int(random(0, 256)), int(random(0, 256)), int(random(0, 256)));

  // Initialize the images
  PImage[] images = new PImage[7];

  images[0] = loadImage("resources/gradient.png");
  images[1] = loadImage("resources/lenna_128px.png");
  images[2] = loadImage("resources/gradient.png");
  images[3] = loadImage("resources/self_portrait.png");
  images[4] = loadImage("resources/stary_night.png");
  images[5] = loadImage("resources/the_potato_eaters.png");
  images[6] = loadImage("resources/wheat_field_with_cypresses.png");

  // Pick a random image
  int randomPick = int(random(0, 7));
  PImage image = images[randomPick];

  // Resize the image and load in the pixels
  float imageScale = 0.1;

  image.resize(int(image.width * imageScale), int(image.height * imageScale));
  image.loadPixels();

  // Calculate the maximum circe diameter (TODO: does not function 100%)
  float circleDiameter;

  if (width/height < image.width/image.height) {
    circleDiameter = width/image.width;
  } else {
```

challenge_1_halftone.pde (vervolg)

```
    circleDiameter = height/image.height;
  }

  // convert the image to an halftone
  for (int x = 0; x < image.width; x++) {
    for (int y = 0; y < image.height; y++) {

      // convert the RGB values to a range between 0 and 1
      colorMode(RGB, 1f);

      int pixel = image.pixels[y * image.width + x];
      float red = red(pixel);
      float green = green(pixel);
      float blue = blue(pixel);

      // Convert RGB to CMYK values (source
          https://www.rapidtables.com/convert/color/rgb-to-cmyk.html)
      float black = 1 - max(red, green, blue);

      // Draw the halftone circles on the screen
      ellipse(x * circleDiameter, y * circleDiameter , circleDiameter *
          black, circleDiameter * black);
    }
  }
}
```